The MMSFORTH Newsletter

MILLER MICROCOMPUTER SERVICES
61 LAKE SHORE ROAD, NATICK, MASS 01760
(617) 653-6136

March-April 1980
Vol.1 No.1

GAD! ANOTHER NEWSLETTER! (Editorial)

Welcome to our new (and overdue!) MMSFORTH NEWSLETTER. We offer it as tbe vehicle for the licensed MMSFORTH user community's questions and suggestions, constructive criticism, and for sharing interesting routines and short programs. MMS is sending the first two issues free of charge. A subscription arrangement will be announced and will become effective with the third issue.

Some ground-rules for the MMSFORTH Newsletter:
1. As MMSFORTH and this Newsletter evolve, early descriptions may require updating. (The current version of MMSFORTH is V 1.8.)
2. Number values mentioned in this Newsletter are in single-precision decimal unless noted otherwise.
3. MMS and authors of articles are not liable for damage and frustration attributable to any imperfections herein.
4. We solicit your contributions for this Newsletter in writing or in MMSFORTH, Electric Pencil or SCRIPSIT on tape or disk.

On the first page of our first Newsletter, MMS acknowledges its gratitude to Charles Moore and Elizabeth Rather of Forth Inc. for developing Forth during the past decade, and for encouraging our development of MMSFORTH as a competent version on the Radio Shack TRS-80. The cooperative spirit shown by the staff of this important Forth company is one reason Forth is a better tool.

One of the best sources of non-TRS-80 FORTH information is the Forth Interest Group, P.O. Box 1105, San Carlos, CA 94070. It costs only $5.00 to get aboard, including a list of available documents and the first six issues of FORTH DIMENSIONS, the FIG Newsletter. We're FIG members and tbink you will enjoy it, too.

Are you appreciating MMSFORTH? At MMS, we hear cries of delight from experienced users and many newcomers, an occasional roar of frustration from someone accustomed to weaker but more strictly developed (read: limited) systems, requests for additional routines and for better documentation. We also get your inputs to correct errors and reduce ambiguities. We're grateful for all. It is clear that Fortb looks as good to most of you as it does to us, that the books aren't yet a tenth as good as tbe best on BASIC, and that MMS is hard-pressed to deal individually with everybody until the better books arrive. For the serious professional user, we offer consulting expertise. For the future, we have our own manual - maybe two - in tbe works. For now, we begin this Newsletter to increase our own correspondence with you and to open up your feedback to each other. This MMSFORTH Newsletter is important to us. We hope you'll value it, too.

-- Dick Miller, Editor 4th Class

FUN & GAMES

BOTTLES AND MUGS:

Here's a quickie, demonstrating a simple use of TRS-80 graphics.

```
: MUGS    0  5 SPACES DO  191 79 32 ECHO ECHO ECHO LOOP CR ;
: BOTTLES 0 2 SPACES DO 189 168 32 ECHO ECHO ECHO LOOP CR ;
10 MUGS Enter           15 BOTTLES Enter
```

DIGI-PONG:

Jim Gerow of the Eastern Mass. MUG turned in our first user contribution, his own real-time graphics game in two tightly packed blocks of MMSFORTH. Jim used the MMSFORTH tape system to create this fine demonstration of the space and speed capabilities of Forth. It's real electronic fly-paper! We expect it will spawn various combinations and permutations from other users. Pong, anyone?

BLOCK : 67

```
( DIGI-PONG  J. GEROW 12/79 ) : TASK ;  3 CONSTANT SKILL
33 LOAD ( LOAD GRAPHICS ROUTINES) : PAUSE SKILL 99 * 0 DO LOOP ;
20 VARIABLE R  20 VARIABLE C  1 VARIABLE RI  1 VARIABLE CI
4 VARIABLE P1 16 VARIABLE P2 12 VARIABLE WIDTH
0 VARIABLE SCORE  1 VARIABLE ROUND
: TCHK   R @ RI @ + 9 < IF RI @ MINUS RI ! THEN ;
: LCHK   C @ CI @ + 4 < IF CI @ MINUS CI ! THEN ;
: RCHK   C @ CI @ + 123 > IF CI @ MINUS CI ! THEN ;
: SINIT  CLS  9 7 DO 128 0 DO J I ESET LOOP LOOP 48 9 DO 4 0 DO
 J I ESET LOOP LOOP 48 9 DO 128 124 DO J I ESET LOOP LOOP
48 46 DO 128 0 DO J I ESET LOOP LOOP 0 22 PTC " D I G I - P O "
" N G" 1 8 PTC " PRESS ENTER KEY TO SERVE"  1 47 PTC " ROUND"
1 37 PTC " SCORE 0   ROUND 1"
15 5 PTC " 1" 15 11 PTC " 2" 15 17 PTC " 3"
15 23 PTC " 4" 15 29 PTC " 5" 15 35 PTC " 6" 15 41 PTC " 7"
15 47 PTC " 8" 15 53 PTC " 9" 15 59 PTC " 0" ;    68 LOAD
```

LIFE:

The Game of Life is a population dynamics simulation invented by John Conway, an Englishman, and first reported in tbe October 1970 Scientific American. The December 1978 issue of Byte Magazine is a rich source of Life articles and references. (We liked "Some Facts of Life" on page 54 the best.)

Single cells, represented on the screen as graphics characters, reproduce, live or die governed by a simple set of rules based on how many direct neighbors abut each cell. More than three neighbors and the cell dies of overcrowding, fewer than two and it dies of loneliness, otherwise it survives in the next generation. A cell will be born in any empty space which has exactly three neighbors.

On Block 53 of your MMSFORTH System, MMS has provided a one-block Life module in Forth. It uses the three-block Doodle screen-graphics module which resides immediately beneath, and this in turn uses the standard graphics module and the basic MMSFORTH vocabulary. The whole operation is loaded automatically from the system diskette menu, by calling LIFE.

When Life has loaded and its menu screen is on the video, press L for the Load mode. You will be prompted for a block number at the bottom right of the video screen. At tbis point enter 62, 63 or 64 to load the desired block.

Some beginning patterns of cells are very long-lived, continuing to grow for many generations. Two of these patterns are supplied on the MMSFORTH System Diskette or Cassette as Blocks 62 and 63.

The MMSFORTH version of Life also allows the user to enter his/her own starting patterns via tbe Doodle module. Block 64 contains a Doodle picture which is short-lived if run with Life.

You now can run Life by pressing G for Go, or you can reverse the white and black areas of the screen with R for Reverse. Note that patterns with a lot of white cells will die out relatively fast. If you have started running Life and wish to stop it to do something different hold down the I (Interrupt) key until the blinking cursor returns in the upper right area of the screen. The Life program only scans the keyboard (to see if you are pressing the I key) when the full screen has been updated and before starting the next screen.

To enter your own pattern note the location of tbe small block in tbe center of the screen (the drawing cursor) and the "compass rose", the small group of numbers in the upper right of the screen. The rose indicates direction: you press tbe number which represents the relative direction you wish to move the drawing cursor. The letter in the center of the rose indicates the mode of action: M for Move, D for Draw, or E for Erase. ROW and COLumn indicate the current position of tbe drawing cursor. The rose numbers correspond to the numbers on tbe TRS-80 numeric keypad. Children (of all ages!) enjoy drawing pictures with tbis Doodle program.

If you wish to erase the screen use C to make it Clear and W to make it White. Reverse can be performed repeatedly.

To save a particularly nice pattern or picture, press S for Save and you will be prompted (as with the Load option) for a block number. This block number is where you will write the contents of the video screen so be sure it is not one which has some important data on it. As delivered, the MMSFORTH disk has 21 free blocks from 66 to 86.

BLOCK : 68

```
: PDRAW  P2 @ P1 @ DO 42 I ESET LOOP ;  ( DIGI-PONG, 2 OF 2 )
: PCLR   P2 @ P1 @ DO 42 I ECLR LOOP    ;
: PGET   ?KEY DUP DUP  IF  48 - DUP 0= IF DROP 10 THEN
  DUP 0 >= OVER 10 <= AND IF 12 * 8 - DUP P1 @ <>
  IF PCLR DUP P1 ! WIDTH @ + P2 ! PDRAW ELSE DROP THEN
  ELSE DROP THEN ELSE DROP THEN ;
: ASCORE 1 ROUND +! 1 52 PTC ROUND ?
  10 R ! 0 BEGIN 1+ PGET 13 = END ABS 123 MOD 4 MAX C ! ;
: PCHK 1 R @ RI @ + 42 > IF  C @ P1 @ >= C @ P2 @ < AND
  IF DROP 0 RI @ MINUS RI ! 1 SCORE +!  1 42 PTC SCORE ?
  ELSE R @ C @ ECLR ASCORE 5000 0 DO LOOP THEN THEN ;
: PONG BEGIN CLS " SKILL LEVEL (1-10)" #IN 1 MAX 10 MIN
  (') SKILL !  0 SCORE ! 0 ROUND ! SINIT PDRAW ASCORE
  BEGIN  PGET DROP PCHK IF R @ C @ ECLR THEN  TCHK  RCHK  LCHK
  RI @ R +! CI @ C +! R @ C @ ESET PAUSE ROUND @ 21 > END
 14 63 PTC CR " PLAY AGAIN" Y/N END ; CLS PONG FORGET TASK DIR
```

SETTING FORTH (for beginners)

PAINT (in colors of white, black, striped, and flashing!):

We are going to create a new routine based on the Forth word, FILL . We will do so by using some immediate-execute mode operations. We also will store the program in an edited block for further modification later.

First, try the word FILL to see bow it performs. In your MMSFORTH Glossary we find tbat FILL expects three numbers on stack and will fill the top number of bytes starting at the address 2nd-on-stack, with the character whose ASCII code is 3rd-on-stack. Use it to paint the screen (whicb starts at RAM location 15360, a good number to remember!) white with ASCII Code 191 :

```
191 15360 1024 FILL  Enter
```

Fast, what? But we can make it faster by defining a single word to do all except the ASCII code, and then we can use various codes to get different displays. Just before we start, put a special word at the beginning of your new words, like a bookmark:

```
: TASK ;
: PAINT  15360 1024 FILL ;
191 PAINT  Enter
```

Get the idea? If you believe tbat, we'll tell you another:

```
: WHITE    191 PAINT ;
: BLACK    128 PAINT ;
: STRIPE   131 PAINT ;
WHITE   Enter      BLACK  Enter    STRIPE  Enter
```

Whoopee! We can even make it flash, like so:

```
: FLASHES   0 DO  WHITE  BLACK  LOOP ;

5 FLASHES  (Enter)
```

Whoops, a problem! Forth is so fast that the flashes are ineffective. We can add a pause routine, but first we will forget the latest word so we can redefine it with the new PAUSE word pre-defined for use therein:

```
FORGET FLASHES
: PAUSE   1000 0 DO  LOOP ;
: FLASHES   0 DO  WHITE PAUSE  BLACK PAUSE PAUSE  LOOP ;
5 FLASHES
```

Why the double pause after BLACK ? Because I liked it better, that's why. Try it yourself. When you've had enough, you can remove these new words from the top of the dictionary and free the RAM for other uses, by saying FORGET TASK . Then use these techniques and your Glossary for other projects.

NOTE: FORGET TASK frees the use of these words, as well as their RAM space. In this example WHITE will supersede WHILE in newer definitions until we FORGET TASK . This is because MMSFORTH only recognizes the first three characters plus the number of characters. (BASIC only saw the first two, remember?) This feature can be used effectively by advanced programmers.

Now to contemplate that which we have created. Using Forth, we quickly and easily added special new words for our own tasks to the existing FORTH vocabulary of the MMSFORTH dictionary. These words are compiled immediately; that is, their machine code definitions are added to the dictionary in RAM and they become as useable as the preceding words. But the source code, the sequence of Forth words we wrote on the screen, is no longer available for modification or to reload tomorrow night. To store it, we need an Editor.

As fate would have it, two fine editors are on the MMSFORTH system tape or disk. One is much as described in the microFORTH PRIMER, and the other is a more powerful full-screen editor built upon the former. Either can be used to create the source code in a Forth block, from which it can be used, modified and stored to tape or disk. I will use tbe fancy version here.

The MMSFORTH block is the same size as your TRS-80's video display screen: 16 lines (numbered 0 through 15) of 64 characters each, adding up to 1024 characters or bytes or, as we say in the byte business, 1K. Your original MMSFORTH tape or disk has 66 blocks of software aboard; it is duplicated on the reverse side of the tape, and there are 21 additional blank blocks on the disk. For now, we will build the program in a temporary block, in the computer's memory. Two block buffers are provided, and the actual block numbers presently "attached" to their contents are the negatives of two variables, BK1 and BK2 . Load the optional full-screen editor from Block 65. Then, taking care to select an available block number, enter the Editor vocabulary, clear that block to blanks, and enter the "edit-screen" mode in the same block:

```
DIR    SCREEN   (on disk) or    65 LOAD   (on tape)
EDITOR  80 CLEAR  V
```

Here is a blank slate on which to create our source code record. Start by titling your work of art on Line 0. Press the R key to Replace the blanks, then write:

(MY FIRST MASTERPIECE!) or some other significant statement. Add : TASK ; near the right end of this line, then press Enter to return from the Replace submode to the Editor mode. Press Enter again to move down to the beginning of the next line. Let's skip a line with another Enter, just to keep things readable and to leave some room for possible future changes. On line 2 (remember, we started on Line 0), we again can enter

```
: PAINT   15360 1024 FILL ;
```

Did you remember to press R on the new line? Other submodes are Insert and Delete. In general, return to the Editor mode with an Enter between each of these. Experiment, then complete the earlier program on this block. These submodes are all you'll need, even to correct mistakes and to make changes.

When you like the block and are ready to save it, from the Editor mode press S to substitute this version for the prior one in the block buffer (or press Q to "forget it" and keep the prior one). Now it is the new Block 80, or whatever (you can get the present screen number with SCR ?). You can again compile from this source code, by entering 80 LOAD . This time the source code will still be available in Block 80 when you are through. Try it out, by entering 5 FLASHES again. But remember that repeated LOADs will compile more and more copies of your source code into the limited amount of RAM and eventually may crasb the system; then you will have to reload Forth, or reenter it with Break Reset SYSTEM Enter /19200 . It is safer and easier to FORGET TASK (the first new word) before reLOADing.

As you get into fancier programming, you will create multiple-block programs and Forth's virtual memory will move the earlier blocks out to disk or tape as they are displaced by new text entering the two block buffers. But for now, and for the final two blocks later, if you want to keep this block move it out to tape or disk yourself with the word, FLUSH . If you'd rather forget the present contents of the two block buffers, say ERASE-CORE to ensure that the virtual memory doesn't flush them for you later. To FLUSH on tape, you must provide the correct place on tape, in Record mode. The disk system overwrites that block number automatically. Be sure you aren't write-protected or, on disk, PBLK'd.

Now you have created you own Forth program, saved its source in a Forth block, and saved that block for use tomorrow. Let's call it a day, and an interesting one at that.

BACKUP:

Some beginners wonder why 0BACKUP permits exchange of diskettes before execution but BACKUP (from Drive 0 to Drive 1) does not. It does. We gave BACKUP equal billing on the directory menu, but in fact it is already aboard in basic MMSFORTH. (Note that it doesn't appear in Block 10 as a directory command.) So just swap diskettes when you wisb, then call BACKUP for immediate action. Remember that your destination diskette must be formatted, un-PBLK'd, and have its write-protect tab removed.

PERIPHERAL TALK

DISK DRIVES:

Here's a Forth routine from MMS, to check your disk drives for alignment and for speed-up capabilities:

```
: 0TEST   BECIN  15360 0 RBLK  15360 86 RBLK  ?KEY END ;
0TEST  Enter   (hold any key to exit)
```

If the above test works without hanging up, and without chattering excessively while failing to read the complete first or last block on your 0-drive diskette, your system is working OK at the usual low speed associated with Radio Shack disk drives. Let's try to speed it up, with one-byte stores of different seek and restore constants, as follows:

```
22 20852 C!     2 20824 C!
```

Now repeat the 0TEST routine. If it works you'll hear the briefer delay between tracks quite clearly. If it fails put back the more conservative original values, which were 23 and 3 respectively. If it works, press your luck by trying new constants of 21 and 1. In descending order, these constant pairs correspond to time delays of 40 milliseconds, 20 ms and 12 ms per track. MMS sells and uses MPI drive mechanisms, which are capable of 5 ms but which we run at 12 ms on the TRS-80. Even many Radio Shack (Shugart SA400) drives are capable of the 20 ms speed-up, so give it a try!

Want to use the disk drive diagnostic on mnltiple drives? Here's a version for that:

```
: TEST   87 *  15360 SWAP  BECIN  OVER OVER RBLK
  OVER OVER 86 + RBLK  ?KEY END  DROP DROP ;
```

Then 2 TEST Enter for a test of Drive 2, etc. Don't forget to move your diskette into that drive. If you get a "PROTECT - DISK ERROR" rejection slip, just set 3 #DV C! like you should have in the first place, then try again!

TAPES:

Several months ago, MMS delivered some tapes of marginal recording quality. At the time, they loaded easily on the TRS-80's at MMS, but several users reported exceptionally hard loading. The fault was finally traced to an improper alignment procedure at the factory service center for our high-speed cassette duplicator - but not until too much time-consuming trouble-shooting and several patience-consuming round trips for alignment. We believe we have corrected all bad tapes, and want to sssnre our cnstomers that the Version 1.8 MMSFORTH Cassette loads very easily. (It's available now as a $10.00 rewrite on your original MMSFORTH System Cassette, plus $1 shipping/handling.) MMS thsnks those customers wbo suffered with us, for their confidence.

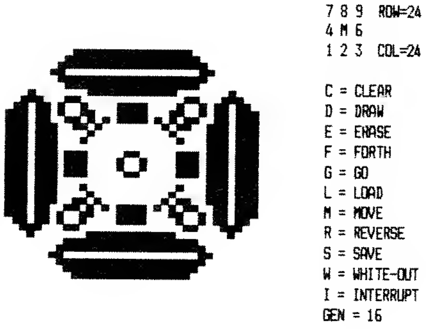
PRINTERS:

In genersl, use PLIST instesd of LIST when going to printer - LIST ends in QUIT, which st the very least will cancel CRT (for example, in PRINT 42 LIST CRT) snd leave you without a video display.

Don't use CLS in PRINT operations - many printers will misinterpret its 31 ECHO to reset character width, etc.

SERIAL PRINTERS and other non-Centronics-port specials are, in general, compatible witb MMSFORTH without custom programming. Just use your existing printer-driver routine. First size s copy of your MMSFORTH System to be smaller than the starting RAM location in your driver. (For a driver wbich starts at 60000, enter: 59999 19203 ! ERASE-CORE 19200 0 0 2 40 DWTSECS .) Then bring up that driver in your usual manner, without Forth; bring up your custom-sized MMSFORTH System; finally, store the driver's entry address in 16422: 60000 16422 ! and you are up and running.

MMS sells snd uses the nifty new Okidata Micro-Line 80 dot-matrix impact printer, which prints upper and lower case plus the actual TRS-80 GRAPHICS CHARACTERS set! If you are using this printer or anotber with similar cspabilities, permit your JKL function to print the graphics cbarscters too, instead of substituting periods as more mundane printers must. Just edit tbe 126 on Block 34, Line 2 to s 191 and you're ready to print some fine Life displays, grapbs, etc. (When Life does its FORCET SCR upon loading, it will forget the JKL words, too - unless you're smart enougb to load JKL afterwards. Just press Break once Life is loaded, 34 LOAD for JKL, then enter LIFE to proceed). For best duplicstion of screen proportions, use 6 lines/inch snd 16.5 cbaracters/inch (on tbe u-80, send 27 ECHO 54 ECHO 29 ECHO . A 27 ECHO 66 ECHO will center tbe display on the page, as well). We like it!



```
7 8 9  ROW=24
4 M 6
1 2 3  COL=24

C = CLEAR
D = DRAW
E = ERASE
F = FORTH
G = GO
L = LOAD
M = MOVE
R = REVERSE
S = SAVE
W = WHITE-OUT
I = INTERRUPT
GEN = 16
```

INSIDE TRACK (for advanced users)

ADUMP:

As home-base on the East Coast for NEWDOS+, MMS apprecistes the top-notch SUPERZAP display of RAM. So altbough your basic MMSFORTH system includes a compact DUMP command, here is a far more elegant ADUMP routine from user Andy Watson aided and abetted by Dick Miller of MMS. ADUMP displays tbe hexadecimsl AND alpbanumeric ASCII versions of the dump simultaneously, along with the hex locations, a complete first line with tbe actual starting position noted witb down-arrows, each new 256-byte "sector" starts with a blank line..... Enjoy!

Once you've loaded ADUMP, let's use it to examine its own code in RAM. Like DUMP, it wants the starting address 2nd-on-stack and the number of bytes of code at top-of-stack. Unlike DUMP, it automatically switches into bex, tben exits back to the number base you were in. ADUMP's first definition is 2DUP and each dictionary entry is preceded by sn 8-byte hesder, so it begins in RAM at:
' 2DUP 8 - 90 ADUMP Enter

The ADUMP display marks the beginning of the 2DUP entry with two down-srrows, one at the number 04 on tbe central HEX display, snother at a period (i.e., no equivslent ASCII character) in the ASCII display. Reading from here, it shows that the word has four chsracters and the first three of tbese are 2DU, gives the present RAM entry address of the preceding word in the dictionary, and the address of the type of word it is (colon-definition, vsriable, code, etc.). Following this header is the word's machine language definition, the psrt to which new words using this one will point. You are examining, live, the Indirect Threaded Code which typifies Forth. Of course, ADUMPs may be sent to printer with PRINT, PCRT or the JKL option.

MEMORY MAP:

The adjacent MMSFORTH Memory Map is written upside-down because:
1. this is compatible with Radio Shack's Level II BASIC memory map;
2. it msdes "top-of-stack" look like it's on the top (the stacks really grow toward lower RAM), and
3. because a lot of Forth is backward, anyway.

Although experimenting with some of the information is strictly for the experts, note that your further programming ares is approximately the space between the top of dictionary and the top (bottom?) of stack. Ask for it witb 'S PAD - or define a word to do the same (how about MEM-SIZE?).

```
Hex   Decimal   Memory Map              Forth Word(s)
                                          for Address
0      0        :=================:
                :    Level II     :
                :    BASIC ROM    :
3000  12288     :=================:
                :    Keyboard     :
                :   & other I/O   :
3C00  15360     :-----------------:
                :    Video RAM    :
4000  16384     :-----------------:
                :  DCB's & misc.  :
4300  17152     :=================:
                : Block Buffers:  :
                :       BK1       :
4700  18176     :-----------------:
                :       BK2       :
4B00  19200     :=================:    (enter Forth)
              D :Forth Dictionary::
              I :Mach.Lang. Kernel:
4C4C  19532   C :-----------------:    ' FORTH 8 -
              T :Forth source code:
              I :  not provided   :
5D63  23907   O :-----------------:    ' OCTAL 8 -
              N :Forth source code:
              A :    provided     :
702C  28716   R :-----------------:
              Y : User mods from  :
                :Forth source code:
                :- - - - ↧ - - - -:    HERE
                :   Word Buffer   :
                :--------↧--------:    PAD (65 above HERE)

                   This is your
                 svailable RAM space
                    for Forth!

              S :- - - - ↥ - - - -:    'S
              T : Parameter (User):
              A :      Stack      :
7F00  32512   C :-----------------:    19207 @ = neg. of RS space
              K :- - - - ↥ - - - -:    R↑ (in Assembler)
              S :  Return Stack   :
7FE0  32736     :-----------------:    19205 @ = neg. of LS spsce
                :- - - - ↥ - - - -:
                :   Load Stack    :
7FFF  32767     :=================:    19203 @  plus 65535
                                         = RAM "ceiling"
```

MMSFORTH V 1.8 MEMORY MAP (16K RAM shown)

```
( ALPHA-HEX DUMP ROUTINE - ALFA FROM 'JKL' ROUTINE )
( FROM ANDY WATSON, WITH MMS MODS. BY A.R.MILLER, 2/8/80 )
CODE 2DUP   HL POP DE POP DE PUSH HL PUSH PSH2
: ALFA   2DUP 0 DO DUP I + C@ DUP 32 < SWAP 126 > +
  IF 46 OVER I + C! THEN LOOP DROP ;
: 1DUMP   0 SWAP 0 DO DUP 15 AND 0= IF CR OVER <# # # # # #>
  TYPE SPACE THEN DUP 1 AND 0= IF SPACE THEN OVER C@
  <# # #S #> TYPE 1+ SWAP 1+ SWAP LOOP DROP DROP ;
: 2DUMP   2 /MOD 5 * SWAP 2 * + 6 + ;
: ADUMP   BASE C@ ROT ROT HEX CR SWAP DUP 16 MOD DUP DUP 2DUMP
  DUP SPACES 92 ECHO 46 SWAP - + SPACES 92 ECHO SWAP OVER -
  ROT ROT + SWAP BECIN <R DUP 16 > IF 16 - 16 SWAP ELSE 0 THEN
  <R <R I 0 > IF J 255 AND 0= IF CR THEN J I 1DUMP 2 SPACES
  J PAD 1+ I MOVE I PAD C! PAD 1+ I ALFA I 1+
  2DUMP 48 SWAP - SPACES TYPE R> DROP R> R> 16 + 0
  ELSE R> DROP R> R> 1 THEN END DROP DROP CR BASE C! ;
```

CET-TOGETHER

The MMSFORTH User Group (MUG) of Eastern Massachusetts has its own newsletter and meets on the third Wednesday evening of each month in Cochituate, Mass. It's a lively mix of beginning and advanced users of tape and disk MMSFORTH. MMS normally attends this meeting and provides some of the talks and demonstrations. Contact: Jim Gerow, 1630 Worcester Road, Framingham MA 01701 (617/872-1882).

We don't know of other existing MMSFORTH Users Groups, but here are some users interested in starting local ones:
Morris Herman (MMSFORTH Dealer), 503 Rosario Drive, Santa Barbara, CA 93110 (805/964-7144)
Paul Van der Eyk, 4910 Fran Place #204, Alexandria, VA 22312 (703/354-7443)

We now have many hundreds of registered users, so add your name to this liat to get a good thing going in your region.

NOTE: Program trading is one popular facet of these meetings, hut NOT commercial programs and WITHOUT MMSFORTH systems ahoard! Promote legitimate sharing, discourage pirating, and take care not to jeopardize your own MMSFORTH serial number.

AT MILLER MICROCOMPUTER SERVICES

WHAT'S COMING:

MMS is very pleased with the early responses regarding our new DATAHANDLER data base management aystem. It is hased on the PIMS introductory text hy SCELBI Puhlications, hut is much more flexihle and powerful. It selects on any field or portion of a field in ahout a half-second, and sorts hundreds of records in 5 to 20 seconds. THE DATAHANDLER V 1.1 (with user-specified one-line reports) is heing released now, and additional modules to add to its capahilities are spinning off our keyboard at this time. This inexpensive, easy to use and very professional DBMS is availahle now for your TRS-80. Yes, it comes with MMSFORTH source code for your own modifications.

In response to many user requests, MMS now offers a very thorough single and douhle precision FLOATING-POINT MATH PACKACE for use with your MMSFORTH System Diskette. In addition to the Level II BASIC capahilities, it has complex numher math, rectangular and polar coordinate conversions, and an optional degrees mode for its trigonometric functions.

A FULL Z80 ASSEMBLER for MMSFORTH is also availahle. It and the Floating-Point Math package are included on a single diskette for $29.95 plus the usual shipping/handling and Mass. tax.

MMSFORTH utilities in development include a VIDEO DISPLAY DRIVER ROUTINE for the new Radio Shack lower-case characters modification, an RS-232-C COMMUNICATIONS PACKACE in MMSFORTH, and a FULL ASCII KEYBOARD. FORTHWRITE, our word-processing program project, was suspended while we awaited Radio Shack's SCRIPSIT hut it appears that there is room for competition after all. Comments regarding this?

SPECIAL APPLICATIONS:

MMS is very husy developing CUSTOM applications in MMSFORTH. We find it ideal for many client purposes, and know that many advanced users are also applying it to serious professional tasks. A hrief item descrihing your successful project will he most welcome.

MMSFORTH MODIFICATIONS

UPGRADE YOUR MMSFORTH SYSTEM TO V 1.8:

If you like, MMS will rewrite your original MMSFORTH System Tape or Disk for only $10.00, plus $1.00 shipping/handling and $4.00 extra which will be refunded with your order unless a new diskette/cassette is reqnired during our duplication process. If you prefer, for the same $10.00 plns $1.00 we will ship you a V 1.8 Upgrade Diskette which you can merge yourself.

Version 1.8 has added quite a bit for some of our early customers: the full-screen editor on Block 65 (including one-key stepping to the adjacent hlocks), intelligent ' and " (that means they work hoth inside and outside of colon-definitions), and many minor improvements.

VERSION 1.8 MMSFORTH GLOSSARY:

This February 1980 edition is now availahle. It costs $4.00 plus $1.00 shipping (or just $1.00 for the two items). In addition to more details, corrections, examples and upgrades, it is a complete glossary instead of the early references to the microFORTH PRIMER, and it is printed in upper and lower case type.

#IN CHANGE:

Effective with MMSFORTH Version 1.8, #IN accepts only single-precision integer numhers. Early copies of the CHECK demo program on the system disk/tape and the CHECK print-out routine on THE DATAHANDLER hoth got into tronble with this one, since their programs expect #IN to absorb a douhle-precision numher. Please note this change in your own programming. All new copies are corrected; if you need the fix and are not getting a Version 1.8 upgrade from MMS at this time, just write us for a print-out of the hlock yon need.

MERCING APPLICATIONS WITH YOUR SYSTEM DISKETTE:

THE DATAHANDLER, our new FLOATING-POINT MATH/Z80 ASSEMBLER package and other similar items are delivered on diskette blocks, without a system ahoard. This permits you to merge them with your own serialized MMSFORTH System Diskette upon receipt. The merged disk should display your system's version number in its first screen, hut instead may incorrectly show the version number MMS used to write the new hlocks. To avoid or correct this minor problem, rewrite disk sectors 0 and 1 FROM your MMSFORTH System diskette TO the other disk. If the disk requires further adjustment of either of these two hoot sectors, which lie nnder Block 0, do that also. We intend to incorporate this slightly modified process in future documentation.

To move the two sectors across, hoot your MMSFORTH System and enter PAD 0 0 0 2 DRDSECS . Then swap your program disk into Drive 0 and enter PAD 0 0 0 2 DWTSECS

THE LAST WORD: "May the FORTH he with you!"
- from Tom Dowling, with whom it indeed is.

Punsters: submit a cornier last word for the next issue!

MMS MILLER MICROCOMPUTER SERVICES
61 LAKE SHORE ROAD, NATICK, MASS. 01760
(617) 653-6136